In the claims:

For the Examiner's convenience, all pending claims are presented below with changes shown in accordance with the mandatory amendment format.

1. (Currently Amended) A chipset comprising:

a graphics accelerator;

a memory controller; and

a queue mechanism divided to include ~~including~~:

a first functional unit block (FUB), coupled to the graphics accelerator, to perform a first set of functions for the queue mechanism; and

a second FUB, coupled to the memory controller, to perform a second set of functions for the queue mechanism.

2. (Original) The chipset of claim 1 wherein the queue mechanism further comprises control logic to facilitate an interface between the graphics accelerator and the memory controller.

3. (Original) The chipset of claim 1 wherein the first FUB is operated based upon a first clock domain and the second FUB is operated according to a second clock domain.

4. (Original) The chipset of claim 1 wherein there is unidirectional signaling between the first FUB and the second FUB, such that there will be a strobe and a packet associated with the strobe that flows from the first FUB to the second FUB.

5. (Original) The chipset of claim 3 wherein the second FUB comprises storage elements in which to store information that is written into the queue mechanism.

6. (Original) The chipset of claim 5 wherein the first FUB comprises:

logic associated with a load pointer, wherein the load pointer indicates a location in the storage elements to store information; and

match logic.

7. (Original) The chipset of claim 6 wherein the second FUB comprises:

an unload pointer to indicate a location in the storage elements in which information is to be read from; and

clock gating elements to gate the load pointer into the second clock domain.

8. (Original) The chipset of claim 7 wherein the match logic compares the load and unload pointer to determine whether information is stored in the queue.

9. (Original) The chipset of claim 8 wherein the load pointer is clock crossed to the second clock domain in FUB 1 to save a clock of latency.

10. (Original) The chipset of claim 9 wherein the unload pointer is clock crossed to the first clock domain in the second FUB.

11. (Original) The chipset of claim 10 wherein data to be stored in the storage elements is directly flopped in the first clock domain within the second FUB.

12. (Original) The chipset of claim 10 wherein the clock crossed versions of the load pointer and the unload pointer are used to determine at the second FUB if a command is present.

13. (Original) The chipset of claim 12 wherein the availability of space in the storage elements is determined at the match logic by using the load pointer and the clock crossed version of the unload pointer.

14. (Currently Amended) A system comprising:

a first component;

a second component; and

a queue mechanism divided to include ~~including~~:

a first functional unit block (FUB), coupled to the first component, to perform a first set of functions for the queue mechanism; and

a second FUB, coupled to the second component, to perform a second set of functions for the queue mechanism.

15. (Original) The system of claim 14 wherein the first FUB is operated based upon a first clock domain and the second FUB is operated according to a second clock domain.

16. (Original) The system of claim 15 wherein the second FUB comprises storage elements in which to store information that is written into the queue mechanism.

17. (Original) The system of claim 16 wherein the first FUB comprises:

logic associated with a load pointer, wherein the load pointer indicates a location in the storage elements to store information; and

match logic.

18. (Original) The system of claim 17 wherein the second FUB comprises:

an unload pointer to indicate a location in the storage elements in which information is to be read from; and

clock gating elements to gate the load pointer into the second clock domain.

19. (Currently Amended) A queue mechanism comprising:

a first functional unit block (FUB), coupled to a first component, to perform a first set of functions for the queue mechanism; and

a second FUB, coupled to a second component, to perform a second set of functions for the queue mechanism.

control logic to facilitate an interface between the first component and the second component.

20. (Original) The queue mechanism of claim 19 wherein the first FUB is operated based upon a first clock domain and the second FUB is operated according to a second clock domain.

21. (Original) The queue mechanism of claim 20 wherein the second FUB comprises storage elements in which to store information that is written into the queue mechanism.

22. (Original) The queue mechanism of claim 21 wherein the first FUB comprises:

logic associated with a load pointer, wherein the load pointer indicates a location in the storage elements to store information; and

match logic.

23. (Original) The queue mechanism of claim 22 wherein the second FUB comprises:

an unload pointer to indicate a location in the storage elements in which information is to be read from; and

clock gating elements to gate the load pointer into the second clock domain.

24. (Original) The queue mechanism of claim 23 wherein the match logic compares the load and unload pointer to determine whether information is stored in the queue.

25. (Original) The queue mechanism of claim 24 wherein the load pointer is clock crossed to the second clock domain in FUB 1 to save a clock of latency.

26. (Original) The queue mechanism of claim 25 wherein the unload pointer is clock crossed to the first clock domain in the second FUB.

27. (Original) The queue mechanism of claim 26 wherein data to be stored in the storage elements is directly flopped in the first clock domain within the second FUB.

28. (Original) The queue mechanism of claim 26 wherein the clock crossed versions of the load pointer and the unload pointer are used to determine at the second FUB if a command is present.

29. (Currently Amended) The queue mechanism ~~chipset~~ of claim 28 wherein the availability of space in the storage elements is determined at the match logic by using the load pointer and the clock crossed version of the unload pointer.

30. (Currently Amended) A computer system comprising:

a memory control hub (MCH) having:

a graphics accelerator;

a memory controller;

a queue mechanism divided to include ~~including~~:

a first functional unit block (FUB), coupled to the graphics accelerator, to perform a first set of functions for the queue mechanism; and

a second FUB, coupled to the memory controller, to perform a second set of functions for the queue mechanism.

~~30~~31. (Currently Amended) The computer system of claim 30 wherein the queue mechanism further comprises control logic to facilitate an interface between the graphics accelerator and the memory controller.

~~31~~32. (Currently Amended) The computer system of claim 30 wherein the first FUB is operated based upon a first clock domain and the second FUB is operated according to a second clock domain.